Remarks

This Application has been carefully reviewed in light of the Office Action mailed July 14, 2005. Applicants appreciate the Examiner's consideration of the Application. Although Applicants believe that all pending claims are allowable over the Examiner's rejections without amendment, Applicants have made a clarifying amendment to independent Claim 20. This amendment is not considered narrowing or necessary for patentability. Applicants respectfully request reconsideration and allowance of all pending claims.

I. Allowable Subject Matter

Applicants note with appreciation the allowance of Claims 7-12, 23, and 25. Pursuant to M.P.E.P. § 1302.14, Applicants respectfully issue a statement commenting on the Examiner's reasons for allowance. Applicants respectfully traverse the Examiner's reasons for allowance to the extent that they are inconsistent with applicable case law, statutes, and regulations. Furthermore, Applicants do not admit to any characterization or limitation of Claims 7-12, 23, and 25 or to any characterization of a reference by the Examiner, particularly any that are inconsistent with the language of the claims considered in their entirety and including all of their constituent limitations.

II. The Office Action Appears to be Incomplete

Applicants respectfully submit that the current Office Action appears to be incomplete. In the Office Action Summary, the Examiner indicates that Claims 1-6 and 13-22 are rejected. The Office Action includes a substantive rejection of Claims 1-2, 4-6, and 13-21 (*See* Office Action, Pages 2-4),[1] in that the Examiner indicates on what basis these claims are rejected. In contrast, the Office Action does not appear to include a substantive rejection of Claims 3 and 22. For example, the Examiner did not indicate in the Office Action the basis for the rejection of Claims 3 and 22, including at least the statutory section (e.g., 35 U.S.C. § 102 or 103) and the one or more references on which the rejection of these claims is based. Since no grounds of rejection were stated or reiterated in the current Office Action with respect to Claims 3 and 22, and since the Office Action does not refer to any substantive rejection from the previous Office Action with respect to Claims 3 and 22,

[1] The Examiner references the previous Office Action for the substantive rejection of Claims 1-2, 4-6, and 15-21.

Applicant respectfully submits that the Office Action is incomplete.

For at least these reasons, Applicant respectfully requests that if the Examiner does not issue a Notice of Allowance based on this Response, the Examiner explains the basis of the rejection of Claims 3 and 22, so that Applicants may consider that explanation in preparing the next Response, if appropriate.

III. The Claims are Allowable over the Rejections Under 35 U.S.C. § 103

In order to establish a *prima facie* case of obviousness, three requirements must be met: (1) there must be some suggestion or motivation, either in the references themselves or in the knowledge available to one skilled in the art, to modify a reference or combine multiple references; (2) there must be a reasonable expectation of success; and (3) the prior art reference (or combination of references) must teach or suggest all of the claim limitations.

A. Independent Claims 1, 4, and 20 are Allowable over the Proposed *Stefaniak-van Eikeren* Combination

The Examiner rejects Claims 1-2, 4-6, and 20-21 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 6,550,054 to Stefaniak, et al. ("*Stefaniak*") in view of U.S. Patent No. 6,618,852 to van Eikeren et al ("*van Eikeren*").[2]

Applicants respectfully submit that the Examiner has not proven a *prima facie* case of obviousness for at least two reasons. First, assuming for the sake of argument that the Examiner has shown the requisite teaching suggestion, or motivation in the cited references to combine or modify *Stefaniak* with *van Eikeren* in the manner the Examiner proposes, the proposed *Stefaniak-van Eikeren* combination still fails to disclose, teach, or suggest each and every element of the claimed invention. Second, Applicants respectfully submit that the Examiner has not shown the requisite teaching, suggestion, or motivation to combine or modify *Stefaniak* with *van Eikeren* in the manner the Examiner proposes.

[2] The Examiner refers to the previous Office Action (mailed December 22, 2004) for the substance of this rejection. (*See* Office Action, Page 2) The Examiner also addresses in the current Office Action certain of Applicants' arguments from the previous Response. (*See* Office Action, Pages 4-6)

1. Independent Claims 1 and 4 are Allowable

Independent Claim 1, for example, recites:

> A method for outputting data from a legacy computer system, the data output in Extensible Markup Language format, the method comprising:
>
> generating a model of the legacy computer system, the model comprising one or more incidents within one or more applications that output data;
>
> mapping the model of the legacy computer system to an Extensible Markup Language schema; and
>
> based at least on the mapping of the model of the legacy computer system to the Extensible Markup Language schema, automatically modifying the one or more applications of the legacy computer system that output data, the one or more modified applications operable to output data written using a Document Object Model from the legacy computer system in Extensible Markup Language.

i. *Stefaniak* does not Disclose, Teach, or Suggest "Based at Least on the Mapping of the Model of the Legacy Computer System to the Extensible Markup Language Schema, Automatically Modifying the One or More Applications of the Legacy Computer System that Output Data . . . ."

At a minimum, *Stefaniak*, whether considered alone or in combination with *van Eikeren*, fails to disclose, teach, or suggest the following limitations as recited in Claim 1:

- based at least on the mapping of the model of the legacy computer system to the Extensible Markup Language schema, automatically modifying the one or more applications of the legacy computer system that output data, the one or more modified applications operable to output data written using a Document Object Model from the legacy computer system in Extensible Markup Language.

Stefaniak discloses the following method for representing terminal-based applications in the Unified Modeling Language: (1) transforming a terminal-based application into an application specification; (2) converting the application specification (not the legacy application) into a modeling language-based representation (e.g., UML); and (3) displaying the modeling language-based representation (of the application specification of the terminal-based application) with a graphical user interface. (*See* Abstract)

The Examiner apparently equates the XML/UML modeling language-based representation of the terminal-based screen application disclosed in *Stefaniak* with

"generating a model of the legacy computer system, the model comprising one or more incidents within one or more applications that output data," as recited in Claim 1. (*See* Previous Office Action, Page 4 citing *Stefaniak*, 1:58-67)[3] Applicants assume that the Examiner is relying on the XML/UML modeling language-based representation of the application specification [generated based on the terminal-based screen application] as allegedly disclosing the "model" recited in the first element of Claim 1. The modeling-based representation disclosed in *Stefaniak* appears to be the result of the entire method disclosed in *Stefaniak*. (*See, e.g., Stefaniak*, Title, Abstract, Field of Invention, Summary of Invention) Applicants will assume, for the sake of argument only, that the Examiner's attempted equation of the modeling language-based representation of the terminal application in *Stefaniak* with the "model" generated in the first element of Claim 1 is possible.

It appears to Applicants that the Examiner later uses this very same XML/UML modeling language-based representation disclosed in *Stefaniak* as allegedly disclosing the one or more modified applications recited in Claim 1. For example, in response to Applicants' argument that *Stefaniak* fails to disclose, teach, or suggest "based at least on the mapping of the model of the legacy computer system to the Extensible Markup Language schema, automatically modifying the one or more applications of the legacy computer system that output data," as recited in Claim 1, the Examiner apparently relies on Column 5, Lines 39-44 of *Stefaniak*. (Office Action, Page 4) In particular, the Examiner quotes the language ". . . end-to-end process flow from a legacy program ***to an XML/UML***" and states, "[W]hen a legacy program is transformed ***to an XML***, the legacy program must be converted/modified, and the elements in the legacy program must be mapped to the Extensible Markup Language schema for the conversion." (Office Action, Page 5; emphasis added)

---

[3] The cited portion of *Stefaniak* discloses, in part, that its invention provides:

> a computer-implemented method that automatically converts text-based screen applications of a legacy computer system into a graphical-based representation thereof. The method includes the steps of transforming a terminal-based screen application into an application specification; ***converting the application specification into a modeling language-based representation***; and, displaying ***the modeling language-based representation*** with a graphical user interface.

(*Stefaniak*, 1:59-67; emphasis added)

First, for at least certain reasons discussed in Applicants' previous Response, Applicants respectfully disagree with the substance of the Examiner's statement. (*See, e.g.*, Previous Response at Pages 11-12)

Second, Applicants respectfully submit that the Examiner's attempt to equate *Stefaniak*'s XML/UML modeling language-based representation with the one or more modified program applications recited in Claim 1 conflicts with the Examiner's attempted equation of *Stefaniak*'s XML/UML modeling language-based representation with the "model" recited in the first element of Claim 1. For example, Claim 1 recites "***based at least on the mapping of the model of the legacy computer system to the Extensible Markup Language schema, automatically modifying the one or more applications of the legacy computer system*** that output data." The Examiner equates *Stefaniak*'s XML/UML modeling language-based representation with the "model" recited in the first element of Claim 1, and equates the flow in *Stefaniak* from a legacy program to the XML/UML modeling language-based representation with "automatically modifying the one or more applications of the legacy computer system [***based at least on the mapping of the model of the legacy computer system to the Extensible Markup Language schema***]," as recited in Claim 1.

Using the Examiner's attempted equations, in order to even possibly meet the limitations recited in Claim 1, the flow in *Stefaniak* from a legacy program to an XML/UML modeling language-based representation would have to be based a mapping of the XML/UML modeling language-based representation to an XML schema. However, this would be impossible because the modeling language-based representation, and thus any mapping of the modeling language-based representation to an XML schema, would not even exist until the flow is completed. Applicants pose the question: How could the flow in *Stefaniak* from the legacy program application to the XML/UML modeling language-based representation be based on a mapping of the modeling language-based representation to an XML schema [as it would have to be to even possibly meet the limitations recited in Claim 1] when the modeling language-based representation [and thus any mapping of the modeling language-based representation to an XML schema] does not yet exist? Applicants respectfully submit that it could not.

ii. *Stefaniak* Merely Discloses Representing Terminal-Based Applications in a Modeling Language Such as UML

In addition, Applicants maintain that *Stefaniak* is entirely unrelated to "***automatically modifying*** [based at least on the mapping of the model of the legacy computer system to the Extensible Markup Language schema] ***the one or more applications of the legacy computer system*** that output data, ***the one or more modified applications operable to output data written using a Document Object Model from the legacy computer system in Extensible Markup Language***," as recited in Claim 1. In *Stefaniak*'s own words, *Stefaniak* relates to "a computer-implemented method for representing terminal-based applications in the Unified Modeling Language, which is useful in the development of business centric applications." (Column 1, Lines 15-18) Merely representing a terminal-based application in the UML simply does not disclose, teach, or suggest "***automatically modifying*** [based at least on the mapping of the model of the legacy computer system to the Extensible Markup Language schema] ***the one or more applications of the legacy computer system*** that output data, ***the one or more modified applications operable to output data written using a Document Object Model from the legacy computer system in Extensible Markup Language***," as recited in Claim 1.

As discussed above, there is no modification of any application in *Stefaniak*; there is merely creation of a model of the legacy applications and then creation of an XML representation ***of that model***. Moreover, **setting aside *Stefaniak's* failure to disclose, teach, or suggest use of a DOM,** there is no modification of any application in *Stefaniak* such that ***the modified application*** is operable to output data from the legacy computer system in Extensible Markup Language. Instead, *Stefaniak* merely discloses "transforming a terminal-based screen application into an application specification," "converting the application specification into a modeling language-based representation" (i.e., the UML model), and "displaying the modeling language based representation [i.e., the UML model] with a graphical user interface." (*Stefaniak*, Abstract and 6:59-62) Even the Title, Abstract, Field of the Invention, and Summary of the Invention sections of *Stefaniak* clearly confirm that *Stefaniak* merely discloses a method for representing terminal-based applications in the unified modeling language, not "automatically modifying the one or more applications of the legacy computer system" such that "the one or more modified applications [are] operable to

output data written using a Document Object Model from the legacy computer system in Extensible Markup Language," as recited in Claim 1.

iii. The Document Object Model and the Proposed *Stefaniak-van Eikeren* Combination

The Examiner acknowledges that *Stefaniak* fails to disclose, in the Examiner's words, "using a Document Object Model." (Office Action, Page 4) However, in the previous Office Action, the Examiner argued that *van Eikeren* teaches "automatically modifying one or more applications of the legacy computer system, the modified application operable to output data written using a Document Object Model from the legacy computer system in Extensible Markup Language." (Previous Office Action, Page 4)[4]

The cited portion of *van Eikeren* (*van Eikeren*, 12:5-10; *see* previous Office Action, Page 4) merely provides its view of what the DOM is and that the DOM may be used (as an example API) to access and manipulate XML data. *Van Eikeren*, however, fails to disclose, teach, or suggest using the DOM in the context of the limitations recited in Claim 1, particularly "automatically modifying [based at least on the mapping of the model of the legacy computer system to the Extensible Markup Language schema] the one or more applications of the legacy computer system that output data, the one or more modified applications operable to output data written using a Document Object Model from the legacy computer system in Extensible Markup Language."

Applicants respectfully submit that the Examiner has not shown the requisite teaching, suggestion, or motivation in either *Stefaniak* or *van Eikeren*, or in the knowledge generally available to one of ordinary skill in the art at the time of Applicants' invention, to combine or modify *Stefaniak* and *van Eikeren* in the manner proposed by the Examiner. Claim 1 is allowable for at least this additional reason.

In Response to Applicants' previous arguments regarding the proposed *Stefaniak-van Eikeren* combination, the Examiner states that *Van Eikeren* is only cited to show the Document Object Model. (Office Action, Page 5) Respectfully, it appears to Applicants that

[4] In the current Office Action, the Examiner clarifies the assertion made in the previous Office Action, stating that "Van Eikeren is only cited to show Document Object Model." (Office Action, Page 5)

the Examiner has merely performed a keyword search on the phrase Document Object Model. Regardless of the fact that the Examiner has found a reference that discloses the Document Object Model, the Examiner must explain why and how, using only prior art references or the knowledge generally available to one of ordinary skill in the art at the time of Applicants' invention,[5] one of ordinary skill in the art at the time of Applicants' invention would have been motivated to modify the particular method for representing terminal-based applications in the UML disclosed in *Stefaniak* with the disclosure of the DOM in *van Eikeren*, or how doing so would purportedly meet the limitations of Claim 1.

In an attempt to rebut Applicants' hindsight reconstruction argument, the Examiner states that "it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the applicant's disclosure, such a reconstruction is proper." (Office Action, Page 6) Applicants respectfully submit that the Examiner's position discounts guidance from the M.P.E.P., as well as governing Federal Circuit case law.

The question raised under 35 U.S.C. § 103 is whether the prior art taken as a whole would suggest the claimed invention taken as a whole to one of ordinary skill in the art at the time of the invention. Accordingly, even if all elements of a claim are disclosed in various prior art references, which is certainly not the case here as discussed above, the claimed invention taken as a whole cannot be said to be obvious without some reason given in the prior art why one of ordinary skill at the time of the invention would have been prompted to modify the teachings of a reference or combine the teachings of multiple references to arrive at the claimed invention. It is clear based at least on the many distinctions discussed above that the proposed *Stefaniak-van Eikeren* combination does not, taken as a whole, suggest the claimed invention, taken as a whole. Applicants respectfully submit that the Examiner has merely pieced together disjointed portions of unrelated references, with the benefit of

[5] If "common knowledge" or "well known" art is relied upon by the Examiner to combine or modify the references, Applicants respectfully request that the Examiner provide a reference pursuant to M.P.E.P. § 2144.03 to support such an argument. If the Examiner relies on personal knowledge to supply the required motivation or suggestion to combine or modify the references, Applicants respectfully request that the Examiner provide an affidavit supporting such facts pursuant to M.P.E.P. § 2144.03.

hindsight using Applicants' claims as a blueprint, in an attempt to reconstruct Applicants' claims.

The M.P.E.P. sets forth the strict legal standard for establishing a *prima facie* case of obviousness based on modification or combination of prior art references. "To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references where combined) must teach or suggest all the claim limitations." M.P.E.P. § 2142, 2143. ***The teaching, suggestion, or motivation for the modification or combination and the reasonable expectation of success must both be found in the prior art and cannot be based on an applicant's disclosure***. *See Id.* (citations omitted). "***Obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either explicitly or implicitly in the references themselves or in the knowledge generally available to one of ordinary skill in the art at the time of the invention***." M.P.E.P. § 2143.01 (emphasis added). ***Even the fact that references can be modified or combined does not render the resultant modification or combination obvious unless the prior art teaches or suggests the desirability of the modification or combination***. *See Id.* (citations omitted). Moreover, "To establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. All words in a claim must be considered in judging the patentability of that claim against the prior art." M.P.E.P. § 2143.03 (citations omitted).

The governing Federal Circuit case law makes this strict legal standard even more clear.[6] According to the Federal Circuit, "a showing of a suggestion, teaching, or motivation to combine or modify prior art references is an essential component of an obviousness holding." *In re Sang-Su Lee*, 277 F.3d 1338, 1343, 61 U.S.P.Q.2d 1430, 1433 (Fed. Cir. 2002) (quoting *Brown & Williamson Tobacco Corp. v. Philip Morris Inc.*, 229 F.3d 1120,

[6] Note M.P.E.P. 2145 X.C. ("The Federal Circuit has produced a number of decisions overturning obviousness rejections due to a lack of suggestion in the prior art of the desirability of combining references.").

1124-25, 56 U.S.P.Q.2d 1456, 1459 (Fed. Cir. 2000)). "Evidence of a suggestion, teaching, or motivation . . . may flow from the prior art references themselves, the knowledge of one of ordinary skill in the art, or, in some cases, the nature of the problem to be solved." *In re Dembiczak*, 175 F.3d 994, 999, 50 U.S.P.Q.2d 1614, 1617 (Fed. Cir. 1999). However, the "range of sources available . . . does not diminish the requirement for actual evidence." *Id.* ***Although a prior art device "may be capable of being modified to run the way the apparatus is claimed, there must be a suggestion or motivation in the reference to do so.***" *In re Mills*, 916 F.2d at 682, 16 U.S.P.Q.2d at 1432 (emphasis added). *See also In re Rouffet,* 149 F.3d 1350, 1357, 47 U.S.P.Q.2d 1453, 1457-58 (Fed. Cir. 1998) (***holding a prima facie case of obviousness not made where the combination of the references taught every element of the claimed invention but did not provide a motivation to combine***); *In Re Jones*, 958 F.2d 347, 351, 21 U.S.P.Q.2d 1941, 1944 (Fed. Cir. 1992) ("Conspicuously missing from this record is any evidence, other than the PTO's speculation (if that can be called evidence) that one of ordinary skill in the herbicidal art would have been motivated to make the modification of the prior art salts necessary to arrive at" the claimed invention.). Even a determination that it would have been obvious to one of ordinary skill in the art at the time of the invention to try the proposed modification or combination is not sufficient to establish a *prima facie* case of obviousness. *See In re Fine*, 837 F.2d 1071, 1075, 5 U.S.P.Q.2d 1596, 1599 (Fed. Cir. 1988).

In addition, the M.P.E.P. and the Federal Circuit repeatedly warn against using an applicant's disclosure as a blueprint to reconstruct the claimed invention. For example, the M.P.E.P. states, "***The tendency to resort to 'hindsight' based upon applicant's disclosure is often difficult to avoid due to the very nature of the examination process. However, impermissible hindsight must be avoided and the legal conclusion must be reached on the basis of the facts gleaned from the prior art.***" M.P.E.P. § 2142 (emphasis added). The governing Federal Circuit cases are equally clear. "A critical step in analyzing the patentability of claims pursuant to [35 U.S.C. § 103] is casting the mind back to the time of invention, to consider the thinking of one of ordinary skill in the art, guided only by the prior art references and the then-accepted wisdom in the field. . . . Close adherence to this methodology is especially important in cases where the very ease with which the invention can be understood may prompt one 'to fall victim to the insidious effect of a hindsight

syndrome ***wherein that which only the invention taught is used against its teacher.***'" *In re Kotzab,* 217 F.3d 1365, 1369, 55 U.S.P.Q.2d 1313, 1316 (Fed. Cir. 2000) (citations omitted; emphasis added). In *In re Kotzab*, the court noted that to prevent the use of hindsight based on the invention to defeat patentability of the invention, this court requires the examiner to show a motivation to combine the references that create the case of obviousness. *See id. See also, e.g., Grain Processing Corp. v. American Maize-Products,* 840 F.2d 902, 907, 5 U.S.P.Q.2d 1788, 1792 (Fed. Cir. 1988). Similarly, in *In re Dembiczak*, the Federal Circuit reversed a finding of obviousness by the Board, ***explaining that the required evidence of such a teaching, suggestion, or motivation is essential to avoid impermissible hindsight reconstruction of an applicant's invention***:

> Our case law makes clear that the best defense against the subtle but powerful attraction of hind-sight obviousness analysis is ***rigorous application of the requirement for a showing of the teaching or motivation to combine prior art references***. Combining prior art references without evidence of such a suggestion, teaching, or motivation simply takes the inventor's disclosure as a blueprint for piecing together the prior art to defeat patentability—the essence of hindsight.

175 F.3d at 999, 50 U.S.P.Q.2d at 1617 (emphasis added) (citations omitted; emphasis added).

With respect to the proposed *Stefaniak-van Eikeren* combination, the Examiner states, "Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to use DOM, as suggested by van Eikeren, to apply to output XML data. The modification would have been obvious because one of ordinary skill in the art would have been motivated to provide a simple means of reading and writing data to and from an XML tree structure." (Previous Office Action, Page 4) The Examiner's purported motivation is a statement from *van Eikeren*. (*see van Eikeren*, 12:8-10) However, this statement would not have in any way motivated one of ordinary skill in the art at the time of invention to combine or modify the system disclosed in *Stefaniak* with the system disclosed in *van Eikeren*. In other words, the Examiner's statement does not provide an explanation as to why it would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the particular method for representing terminal-based applications in the UML disclosed in *Stefaniak* with the disclosure of the DOM in *van Eikeren*, how one of

ordinary skill in the art at the time of Applicants' invention would have done so, or how doing so would purportedly meet the limitations of Claim 1.

It appears to Applicants that the Examiner simply found DOM in a reference and states that it would have been obvious to use DOM with *Stefaniak*. The rules and case law make clear that this is exactly the kind of "hindsight reasoning" that the requirement to show a teaching, suggestion, or motivation in the cited references is designed to prohibit. Applicants respectfully direct the Examiner's attention to the above-cited standard for proving a *prima facie* case of obviousness, particularly to the emphasized portions of the standard. Applicants respectfully submit that the Examiner has not met this burden merely by asserting that certain teachings were allegedly known at the time of Applicants' invention.

It certainly would not have been obvious to one of ordinary skill in the art at the time of invention to even attempt to, let alone to actually, modify the particular method for representing terminal-based applications in the unified modeling language disclosed in *Stefaniak* in the manner proposed by the Examiner. Applicants respectfully submit that the Examiner's attempt to modify *Stefaniak* appears to constitute the type of impermissible hindsight reconstruction of Applicants' claims, using Applicants' claims as a blueprint, that is specifically prohibited by the M.P.E.P. and governing Federal Circuit cases.

Accordingly, since the prior art fails to provide the required teaching, suggestion, or motivation to combine *van Eikeren* with *Stefaniak* in the manner the Examiner proposes, Applicants respectfully submit that the Examiner's conclusions set forth in the Office Action fall well short of the requirements set forth in the M.P.E.P. and the governing Federal Circuit case law for demonstrating a *prima facie* case of obviousness. Thus, Applicants maintain that the Examiner's proposed combination of *van Eikeren* with *Stefaniak* appears to be merely an attempt to reconstruct Applicants' claims, with the benefit of hindsight using Applicants' claims as a blueprint, and is unsupported by the teachings of *van Eikeren* and *Stefaniak*. Applicants respectfully submit that the rejection must therefore be withdrawn.

iv. Conclusions with Respect to Claims 1 and 4

As demonstrated above, Applicants respectfully submit that *Stefaniak* is wholly inadequate as a reference against independent Claim 1. Thus, even if *van Eikeren* did disclose the portions of Claim 1 that the Examiner asserts it discloses, and even assuming for the sake of argument that there was the required teaching, suggestion, or motivation to combine *Stefaniak* with *van Eikeren* as the Examiner proposes, the proposed *Stefaniak-van Eikeren* combination would still fail to disclose, teach, or suggest the limitations specifically recited in independent Claim 1, as is required under the M.P.E.P. and the governing Federal Circuit cases for a *prima facie* case of obviousness.[7]

For at least these reasons, Applicants respectfully request reconsideration and allowance of independent Claim 1 and its dependent claims. For at least certain reasons analogous to those discussed above with reference to independent Claim 1, Applicants respectfully request reconsideration and allowance of independent Claim 4 and its dependent claims.

2. Independent Claim 20 is Allowable

At a minimum, *Stefaniak*, whether considered alone or in combination with *van Eikeren*, fails to disclose, teach, or suggest the following limitations recited in Claim 20, as amended:

- modifying an application of the legacy computer system such that the modified application is operable to output data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application;
- outputting data from the modified application, the output data having the schema element of the target Extensible Markup Language schema;
- aligning the schema element of the output data and a current context;
- writing the schema element of the output data to a current one of plural contexts of the target Extensible Markup Language schema; and

[7] In response to the Examiner's assertion that one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references (*see* Office Action, Page 6), Applicants respectfully submit that they have in fact attacked the Examiner's proposed combination. In particular, Applicants have demonstrated that the proposed combination of *Stefaniak* and *van Eikeren*, including the proposed combination of their individual teachings, fails to disclose, teach, or suggest limitations recited in Applicants' claims as is required for the rejection to be properly maintained. Additionally, Applicants' have demonstrated that the Examiner has not shown the requisite teaching, suggestion, or motivation to combine or modify these references in the manner proposed by the Examiner.

- populating a Document Object Model with the output data to output an Extensible Markup Language instance.

For example, *Stefaniak* fails to disclose, teach, or suggest "modifying an application of the legacy computer system such that the modified application is operable to output data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application," as recited in Claim 20. *Stefaniak* is directed to a method for representing terminal-based applications in the UML. (*Stefaniak*, 1:15-17) In particular, *Stefaniak* merely discloses generating models of legacy applications - it is not modifying legacy computer applications, let alone "modifying an application of the legacy computer system such that the modified application is operable to output data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application," as recited in Claim 20.

The Examiner continues to refer to Figure 6 of *Stefaniak* as allegedly disclosing this element of Claim 20 prior to the amendments presented in this Response. (*See* Previous Office Action, Page 8; Office Action, Page 7) Figure 6 of *Stefaniak* merely discloses "a flowchart depicting a process of generating an XML file representation of the UML model created by the process described in FIG. 5A through 5C [of *Stefaniak*]." (*Stefaniak*, 6:59-62) The UML model is a model of an application specification that was created from a terminal-based application. Thus, Figure 6 of *Stefaniak* discloses generating an XML file representation of a UML model of an application specification, the application specification having been created from a terminal-based application (which the Examiner equates with the legacy computer system application). Figure 6 of *Stefaniak* clearly fails to disclose, teach, or suggest "modifying an application of the legacy computer system such that the modified application is operable to output data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application," as recited in Claim 20. In other words, the terminal-based application disclosed in *Stefaniak* is not modified such that it outputs data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application, as recited in Claim 20.

Additionally, *Stefaniak* discloses a system that describes legacy application screens in terms of a terminal application specification and converts the specification into a UML model. (*See* Abstract and Column 1, Lines 57-67) *Stefaniak* repeatedly teaches that the output of the system is a representation or model of the terminal-based application – there is no modification of the terminal-based application itself in *Stefaniak*. (*See* Title; Abstract; Column 1, Lines 15-18 and 28-31) This is further suggested by *Stefaniak*'s continued use of UML, a modeling language, for representing or modeling – as opposed to modifying – the terminal-based application. In other words, even if "the terminal-based application" in *Stefaniak* is comparable to the legacy computer system applications of Claim 20 (which Applicants do not concede), *Stefaniak* fails to disclose, teach, or suggest "modifying an application of the legacy computer system such that the modified application is operable to output data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application" as recited, in part, in Claim 20.

Stefaniak merely discloses creation of a model of the legacy applications and then creation of an XML representation of that model. *Stefaniak* discloses "transforming a terminal-based screen application into an application specification," "converting the application specification into a modeling language-based representation" (i.e., the UML model), and "displaying the modeling language based representation [i.e., the UML model] with a graphical user interface." (Abstract) *Stefaniak* merely discloses creating a UML/XML reference model of the UML model (*see Stefaniak*, 6:59-62); the system disclosed in *Stefaniak* does not modify any application such that the modified application outputs data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application, as recited in Claim 20. Even the Title, Abstract, Field of the Invention, and Summary of the Invention sections of *Stefaniak* clearly confirm that *Stefaniak* merely discloses a method for representing terminal-based applications in the unified modeling language, not "modifying an application of the legacy computer system" such that the modified application is operable to output "data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application," as recited in Claim 20.

As another example, *Stefaniak* fails to disclose, teach, or suggest "outputting data from the modified application, the output data having the schema element of the target Extensible Markup Language schema," as recited in Claim 20. First, at least because *Stefaniak* fails to disclose, teach, or suggest "modifying an application of the legacy computer system to output data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application," as recited in Claim 20, *Stefaniak* necessarily fails to disclose, teach, or suggest "outputting data from the modified application, the output data having the schema element of the target Extensible Markup Language schema," as recited in Claim 20. Second, *Stefaniak* merely discloses generating an XML file representation of a UML model of an application specification created from a terminal-based application. The terminal-based application (which the Examiner equates with the legacy computer system application recited in Claim 20) is not modified to output data having a schema element of an XML schema and thus, does not disclose, teach, or suggest "outputting data from the modified application, the output data having the schema element of the target Extensible Markup Language schema," as recited in Claim 20.

Moreover, even assuming for the sake of argument only that the application specification disclosed in *Stefaniak* could be equated with the modified legacy computer system application recited in Claim 20 (with which Applicants disagree), the application specification does not "output data having a schema element of a target Extensible Markup Language schema, the output data corresponding to a write operation of the application," as recited in Claim 20. Instead, another component of the system disclosed in *Stefaniak* creates a UML model of the application specification and then generates an XML file representation of the UML model of the application specification.

Applicants respectfully submit that *Stefaniak* fails to disclose, teach, or suggest at least certain of the remaining limitations of Claim 20; however, to avoid burdening the record and in view of the clear distinctions discussed above, Applicants do not specifically discuss each of these distinctions in this Response.

The Examiner acknowledges that *Stefaniak* fails to disclose, in the Examiner's words, "using a Document Object Model." (previous Office Action, Page 9) However, the Examiner argues that *van Eikeren* teaches "populating a Document Object Model with the output data to output an Extensible Markup Language instance." (previous Office Action, Page 9) Applicants respectfully disagree.

As discussed above with reference to Claim 1, the cited portion of *van Eikeren* (Column 12, Lines 5-10; *see* Previous Office Action, Page 4) merely provides its view of what the DOM is and that the DOM may be used (as an example API) to access and manipulate XML data. *Van Eikeren*, however, fails to disclose, teach, or suggest "populating a Document Object Model with the output data to output an Extensible Markup Language instance," as recited in Claim 20. Thus, *van Eikeren* clearly fails to make up for at least this deficiency of *Stefaniak*.

Moreover, as Applicants demonstrated above, the Examiner has not shown the requisite teaching, suggestion, or motivation in the either *Stefaniak* or *van Eikeren*, or in the knowledge generally available to one of ordinary skill in the art to combine or modify *Stefaniak* and *van Eikeren* in the manner proposed by the Examiner. Thus, Applicants respectfully submit that the Examiner's rejection based on the proposed *Stefaniak-van Eikeren* combination does not support a *prima facie* case of obviousness, as is required under the M.P.E.P. and governing Federal Circuit cases. Claim 20 is allowable for at least this additional reason.

For at least these reasons, Applicants respectfully request reconsideration and allowance of independent Claim 20 and its dependent claims.

B. Independent Claim 13 is Allowable over the Proposed *Lection-Stefaniak-van Eikeren* Combination

The Examiner rejects Claims 13-14 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 6,418,446 to Lection, et al. ("*Lection*") in view of *Stefaniak* and further in view of *van Eikeren*. Applicants respectfully disagree and discuss independent Claim 13 as an example.

Lection, whether considered alone or in combination with *Stefaniak*, fails to disclose, teach, or suggest at least the following limitations as recited in Claim 13:

- a computer system having an application that outputs data, each data output instance corresponding to a write operation of the application;
- a writer engine loaded on the computer system and interfaced with the application, the writer engine having an Extensible Markup Language schema as a data file and the writer engine operable to write the data output by the application in plural active contexts;
- wherein the application calls the writer engine when the application outputs data, the writer engine operable to build a Document Object Model instance for output of the data in accordance with the Extensible Markup Language schema.

Lection does not even relate to "outputting data from a Document Object model as Extensible Markup Language," as recited in Claim 13.[8] Instead, *Lection* is directed to a process, system, and method for gathering data having dynamically variable record formats such as those created when a dynamic schema is used with a data repository. (Column 3, Lines 30-34) In other words, *Lection* is directed to mapping ***stored data records, which are already structured in XML format***, to variable record formats. The Examiner appears to equate "the source data" of *Lection*, which is stored as a structure data record, with "the output data" in Claim 13. (*See* Office Action, Page 2-3) Applicants respectfully submit that *Lection* does not support this interpretation. The source data disclosed in *Lection* is merely ***a stored structured data record, it is not data output from an application running on a computer system, and it does not correspond to a write operation of the application*** as recited in Claim 13. Thus, *Lection* fails to disclose, teach, or suggest "a computer system having an application that outputs data, each data output instance corresponding to a write operation of the application," as recited in Claim 13.

[8] In response to Applicants' previous Response, the Examiner simply modified the rejection to include *van Eikeren* as allegedly disclosing a Document Object Model. (*See* Office Action, Pages 3 and 7) First, Applicants respectfully submit that the Examiner has not demonstrated the requisite teaching, suggestion, or motivation in the cited references or in the knowledge generally available to one of ordinary skill in the art to combine these references in the manner the Examiner proposes. Second, the Examiner apparently ignored many of Applicants' other arguments. The Examiner did not provide any response to those arguments, which have been reproduced in this Response. "Where the applicant traverses any rejection, the examiner should, if he or she repeats the rejection, take note of the applicant's argument and ***answer the substance of it.***" M.P.E.P. § 707.07(f) (emphasis added). Applicants respectfully submit that the Examiner has not answered the substance of Applicants' arguments with respect to the allowability of the rejected claims over the references.

The Examiner correctly acknowledges that *Lection* fails to disclose a writer engine, as recited in Claim 13. However, the Examiner argues that *Stefaniak* teaches, as stated by the Examiner, "an engine operable to write that data output by the application in plural active contexts; wherein the application calls the writer engine when the application outputs data, the writer engine operable to build a Document Object Model instance for output of the data in accordance with the Extensible Markup Language schema." (Office Action, Pages 2-3) Applicants respectfully disagree.

As purportedly disclosing these limitations, the Examiner references the following portion of *Stefaniak*:

> The terminal screens are discovered using the transform navigator 19, which produces application and screen specifications 32. The application and screen specifications 32 are then applied to the file warehouse 21, which produces the project file reference model 27. The model 27 is applied to the terminal-to-XML 20, which produces a UML model 34 in a MOF compliant repository. The UML model is then applied to the UML model to XMI/UML DTD generator 22, which produces XMI/UML DTD streams 35. The streams 35 may be used for several purposes, including transmitting legacy screen based application specifications over a network to modeling tools. From the modeling tools the application specifications could be viewed in an object oriented way compliant with the UML standard.

(*Stefaniak*, 5:43-57; Office Action, Pages 2-3)

However, nowhere does this cited portion disclose, teach, or suggest "a writer engine loaded on the computer system and interfaced with the application, the writer engine having an Extensible Markup Language schema as a data file and the writer engine operable to write the data output by the application in plural active contexts" or "wherein the application calls the writer engine when the application outputs data, the writer engine operable to build a Document Object Model instance for output of the data in accordance with the Extensible Markup Language schema," as recited in Claim 13. Instead, the cited portion discloses that a transform navigator creates application and screen specifications of a terminal-based application. Then, a project file reference model is created from the application and screen specifications. Next, a terminal-to-XML component creates a UML model of the project file reference model (created from the application and screen specifications). Next, an XMI/UML DTD generator produces *from the UML model* XMI/UML DTD streams.

Apparently, the Examiner equates the XMI/UML DTD generator with the writer engine recited in Claim 13. (Office Action, Pages 2-3) However, the XMI/UML DTD generator merely generates UML/XML representations of the UML model. The cited portion of *Stefaniak* does not disclose, teach, or suggest that the XMI/UML DTD generator "ha[s] an Extensible Markup Language schema as a data file" or is operable to "write the data output by the application in plural active contexts," as recited in Claim 13.

Additionally, neither *Stefaniak* nor *Lection* discloses, teaches, or suggests "wherein the application calls the writer engine when the application outputs data, the writer engine operable to build a Document Object Model instance for output of the data in accordance with the Extensible Markup Language schema," as recited in Claim 13. First, *Stefaniak* fails to even mention the Document Object Model. Thus, *Stefaniak* (which is the only reference cited by the Examiner as allegedly disclosing this limitation) necessarily fails to disclose, teach, or suggest "the writer engine operable to build a Document Object Model instance for output of the data in accordance with the Extensible Markup Language schema," as recited in Claim 13. Furthermore, neither *Stefaniak* nor *Lection* discloses, teaches, or suggests that an application that calls the writer engine when the application outputs data, as recited in Claim 13.

As Applicants demonstrated above, even the combination of *Lection* with *Stefaniak* fails to disclose, teach, or suggest each and every limitation recited in Claim 13. Moreover, Applicants respectfully submit that the Examiner has not shown the requisite teaching, suggestion, or motivation in the either *Stefaniak* or *Lection*, or in the knowledge generally available to one of ordinary skill in the art at the time of Applicants' invention, to combine or modify *Stefaniak* and *Lection* in the manner proposed by the Examiner. Claim 13 is allowable for at least this additional reason.

With respect to the proposed *Lection-Stefaniak* combination, the Examiner states, "Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate XMI/UML DTD generator, as suggested by Stefaniak into the system of Lection, to produce XMI/UML DTD streams. The modification would have

been obvious because one of ordinary skill in the art would have been motivated to provide a simple means to generate XML codes." (Office Action, Pages 2-3) This statement would not have in any way motivated one of ordinary skill in the art at the time of invention to combine or modify the system disclosed in *Stefaniak* with the system disclosed in *Lection*.[9] In other words, the Examiner's statement does not provide an explanation as to why it would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the particular method for representing terminal-based applications in the unified modeling language disclosed in *Stefaniak* with the particular system and techniques disclosed in *Lection*, or how doing so would purportedly meet the limitations of Claim 13. It certainly would not have been obvious to one of ordinary skill in the art at the time of invention to even attempt to, let alone to actually, modify the particular method for representing terminal-based applications in the unified modeling language disclosed in *Stefaniak* in the manner proposed by the Examiner. Applicants respectfully submit that the Examiner's attempt to modify *Stefaniak* appears to constitute the type of impermissible hindsight reconstruction of Applicants' claims that is specifically prohibited by the M.P.E.P. and governing Federal Circuit cases.

Accordingly, since the references fail to provide the required teaching, suggestion, or motivation to modify *Stefaniak* in the manner the Examiner proposes, Applicants respectfully submit that the Examiner's conclusions set forth in the Office Action fall short of the requirements set forth in the M.P.E.P. and the governing Federal Circuit case law for demonstrating a *prima facie* case of obviousness. Thus, Applicants respectfully submits that the Examiner's proposed modifications to *Stefaniak* (and the proposed combination of *Lection* with *Stefaniak*) appears to be merely an attempt, with the benefit of hindsight, to reconstruct Applicants' claims and is unsupported by the teachings of *Stefaniak* and *Lection*.

Applicants reiterate the heavy burden incumbent on the Examiner for demonstrating a *prima facie* case of obviousness. As illustrated above, the Examiner's rejection based on the

[9] If "common knowledge" or "well known" art is relied upon by the Examiner to combine or modify the references, Applicants respectfully request that the Examiner provide a reference pursuant to M.P.E.P. § 2144.03 to support such an argument. If the Examiner relies on personal knowledge to supply the required motivation or suggestion to combine or modify the references, Applicants respectfully request that the Examiner provide an affidavit supporting such facts pursuant to M.P.E.P. § 2144.03.

proposed *Lection-Stefaniak* combination does not support a *prima facie* case of obviousness, as is required under the M.P.E.P. and governing Federal Circuit cases.

For at least these reasons, Applicants respectfully request reconsideration and allowance of independent Claim 13 and its dependent claims.

C. Rejected Dependent Claims 15-19 are Allowable

The Examiner rejects Claims 15-18 under 35 U.S.C. § 103(a) as being unpatentable over *Lection* in view of *Stefaniak* and *van Eikeren* and further in view of Shanmugasundaram, et al. "Relational Databases for Querying XML Documents: Limitations and Opportunities." Proceedings of the 25th VLDB Conference, Edinburgh, Scotland, 1999 ("*Shanmugasundaram*"). The Examiner rejects Claim 19 under 35 U.S.C. § 103(a) as being unpatentable over *Lection* in view of *Stefaniak* and *van Eikeren* and further in view of *Shanmugasundaram*, and U.S. Patent 6,209,124 to Vermeire, et al. ("*Vermeire*").

Dependent Claims 15-19 depend from independent Claim 13, which Applicants have shown above to be allowable, and are allowable for at least this reason. Additionally, dependent Claims 15-19 recite further patentable distinctions over the references cited in the Examiner's rejections. To avoid burdening the record and in view of the clear allowability of independent Claim 13, Applicants do not specifically discuss these distinctions in this Response; however, Applicants reserve the right to discuss these distinctions in a future Response or on Appeal, if appropriate. Furthermore, with respect to the proposed combinations of references made by the Examiner, Applicants do not admit that the proposed combinations of references are possible or that the Examiner has demonstrated the requisite teaching, suggestion, or motivation in the cited references or in the knowledge generally available to one of ordinary skill in the art the time of Applicants' invention to combine or modify these references in the manner proposed.

For at least these reasons, Applicants respectfully request reconsideration and allowance of Claims 15-19.

IV. No Waiver

All of Applicants' arguments and amendments are without prejudice or disclaimer. Additionally, Applicants have merely discussed example distinctions from the various references cited by the Examiner. Other distinctions may exist, and Applicants reserve the right to discuss these additional distinctions in a later Response or on Appeal, if appropriate. By not responding to additional statements made by the Examiner, Applicants do not acquiesce to the Examiner's additional statements. The example distinctions discussed by Applicants are sufficient to overcome the Examiner's rejections.

Conclusion

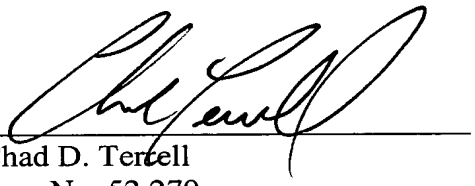
Applicants have made an earnest attempt to place this case in condition for immediate allowance. For at least the foregoing reasons, Applicants respectfully request allowance of all pending claims.

If the Examiner feels that prosecution of the present Application may be advanced in any way by a telephone conference, the Examiner is invited to contact the undersigned attorney at 214.953.6813.

Although no fees are believed to be due, the Commissioner is hereby authorized to charge any additional fees or to credit any overpayment to Deposit Account No. 05-0765 of Electronic Data Systems Corporation.

Respectfully submitted,

BAKER BOTTS L.L.P.
Attorneys for Applicants



Chad D. Tertell
Reg. No. 52,279

Date: October 14, 2005

Customer Number: **35005**